18.1. Introduction

The measurement and analysis of latent variables is of great importance in clinical research. The use of patient reported outcomes in clinical research and epidemiology has increased during the past decades. Such increase was partly due to the phenomenal development of hardware and software in computing. Computer programs nowadays can analyze large data sets of more complexity.

In the recent past, general statistical packages did not allow users to perform item analysis or to estimate parameters of most common models in this specific field. The practitioners were obliged to use various specific software to analyze measurement properties of a questionnaire. Moreover, each of these were often very limited, and often practitioners were obliged to use several software packages to perform a valid analysis, because no single package contained all necessary analyses.

Well-known statistical packages, such as Statistical Analysis System (SAS), SPSS, S-plus R and Stata, have recently (since the end of the 1990s) developed procedures that allow estimating the parameters of linear or generalized linear fixed or mixed models. This very late appearance is certainly due to the increasing complexity of algorithms used in such fields.

In this chapter, we first present software packages for Rasch analysis and describe briefly what each package can do. The reader can find more information about each software package on the Internet. Next, we present and discuss a SAS macro and an R package that programmers in SAS or R language can use. Finally, we present simple
SAS programs that can be used to estimate Rasch or partial credit model parameters, or to generate Rasch or partial credit items.

18.2. Stand alone softwares packages

This section briefly presents stand-alone Rasch software. The list is far from complete, and a more complete list, including direct links to each software Website, can be found at [www.rasch.org/software.html](http://www.rasch.org/software.html). All of the discussed packages provide estimates of item and person parameters, and fit statistics. They differ with respect to the inference frame. DIGRAM is free software, while the others are proprietary.

18.2.1. WINSTEPS

WINSTEPS is a widely used Rasch software program for the windows platform. It uses the joint maximum likelihood (JML) method of estimation of parameters. Originally developed at the University of Chicago, its latest update is from 2009 by Linacre [LIN 09]. In addition to the standard Rasch variable maps, person and item measures, and fit statistics, WINSTEPS also provides many advanced analytical features, such as distractor analysis, principal component analysis of Rasch residuals for assessing unidimensionality and analysis of differential item functioning. It reports tables, files, plots and graphs. A free student version is also available.

18.2.2. RUMM

RUMM was developed at the RUMM Laboratory in Perth, Australia. A time-limited version is usually provided as part of a popular online course offered periodically by several universities. RUMM provides an all-graphical interface that uses familiar point and click set-up features, as found in commercial statistical packages such as SPSS and JMP. It can output a wide range of multicolor charts and graphs, and fit a variety of Rasch models, including dichotomous, rating scale and partial credit models. It is also highly interactive, providing easy reanalysis following diagnosis of previous analyses. The latest version is RUMM2030 [AND 10]. The estimation is based on pairwise conditional maximum likelihood.

18.2.3. CONQUEST

CONQUEST was developed by Wu et al. [WU 07] at the Australian Council for Educational Research as an improved update of Quest, an older software program. Besides the standard dichotomous, rating scale, partial credit and many-facet Rasch
models, CONQUEST also estimates a multidimensional Rasch model using MML methods. It outputs a wide range of informative graphs, charts and variable maps. It is a powerful and flexible program that combines Rasch with latent regression model.

18.2.4. DIGRAM

DIGRAM is part of a larger statistical package, SCD, supporting Statistical analysis of Categorical Data. The program dates back to 1980s and is dedicated to analysis of high-dimensional contingency tables by block recursive chain graphical models. It has since then been expanded to cover other types of models and problems as well, while still focusing on problems that could be solved complete or partly by simple tests for conditional independence. The current version of DIGRAM supports item analysis by graphical and log-linear Rasch models for dichotomous and polytomous items using the conditional frame of inference. Importantly, DIGRAM provides exactly the same tests of fit for the GLLRM as for the pure Rasch model. DIGRAM also provides a number of routines that can be used to construct a GLLRM using an item screening procedure [KRE 11]. DIGRAM is free software, but is not as well-known as the three previously discussed software packages.

18.3. Implementations in standard software

Recently, software procedures, such as NLMIXED in SAS, the NLME library in Splus, and GLLAMM in STATA, have made nonlinear random effects models like the Rasch model in the marginal frame of inference, which is a standard tool. On the other hand the theoretical relationship between the Rasch model and the multiplicative Poisson model has been well-known for many years. This means that conditional maximum likelihood estimation can be done using generalized linear models [TJU 82, KEL 84] and can thus be implemented in standard statistical software.

The parameters of the Rasch model can thus be estimated in standard software like SAS [CHR 06]. This section describes software using these possibilities.

18.3.1. SAS macro for MML estimation

The SAS macro %ANAQOL was written by Hardouin and Mesbah [HAR 07]. It easily and automatically provides (1) indices like Cronbach coefficient $\alpha$, (2) various useful graphical representations, including different kinds of item traces and Backward Reliability Curves, and (3) estimation of the parameters of five IRT models among the most famous (Rasch, Birnbaum, OPLM, partial credit model (PCM) and
rating scale model (RSM)). Goodness of fit tests are implemented for dichotomous items, and the fit of the models can be evaluated by graphical comparison of the expected and observed item characteristics curves, as well as infit and outfit test statistics. Results are composed of outputs of SAS procedures, specific tables and graphical representations.

18.3.2. SAS macros based on CML estimation

Christensen and Bjorner [CHR 03] describe a number of SAS macros that can be used for latent variable models based on the Rasch model. They present macros for item parameters estimation using conditional maximum likelihood (CML), for testing the assumption of unidimensionality and for fitting regression models where either outcome variables or covariates are latent variables measured using (loglinear) Rasch models.

Most notably, their macro %RASCH can do CML estimation for dichotomous and polytomous items. Plots of observed and expected item category frequencies stratified by the total score are also available. The macro %LLRASCH fits some simple log-linear Rasch models and the Martin-Löf test described in Chapter 9 is implemented in their macro %PML. Christensen and Bjorner also present macros for latent regression.

18.3.3. eRm: an R Package

The eRm package (http://epub.wu.ac.at/332/) fits the following models: the Rasch model, the RSM and the PCM, as well as linear reparameterizations through covariate structures, such as the linear logistic test model (LLTM), the linear rating scale model (LRSM) and the linear partial credit model (LPCM). It uses an unitary, efficient CML approach to estimate the item parameters and their standard errors. Graphical and numerical tools for assessing goodness of fit are provided.

18.4. Fitting the Rasch model in SAS

Finally, we present a few simple programs illustrating how to (1) simulate a data set of 10 dichotomous Rasch items and (2) estimate, using different methods, their item parameters.

18.4.1. Simulation of Rasch dichotomous items

Responses to a set of Rasch dichotomous items can be simulated using the following program:
The purpose of the first data step is to fix the sample size $N$ and simulate the latent variable with a normal distribution with zero mean and unit variance. The role of the second data step is to generate a sample of $N$ responses of 10 binary Rasch items, with specified values for item parameters. We have chosen item parameters located in increasing positions on the normal scale, but other values can be chosen. We can, then, directly put the chosen value of the item parameter.

### 18.4.2. MML estimation using PROC NLMIXED

MML estimation using PROC NLMIXED is done as follows:

```sas
/* simulate theta */
%let N=1000;
data latent;
do ind=1 to &N;
   theta=rannor(0);
   output;
end;
run;
/* simulate item responses */
data Rasch;
   set latent;
   bi=quantile('NORMAL',0.05);
   b2=quantile('NORMAL',0.15);
   
   b10=quantile('NORMAL',0.95);
   L1=bi+theta; E1=exp(L1)/(1+exp(L1)); X1=rabin(0,1,E1);
   L2=b2+theta; E2=exp(L2)/(1+exp(L2)); X2=rabin(0,1,E2);
   
   L10=b10+theta; E10=exp(L10)/(1+exp(L10)); X10=rabin(0,1,E10);
   total=X1+X2+X3+X4+X5+X6+X7+X8+X9+X10;
run;
```

The purpose of the first data step is to fix the sample size $N$ and simulate the latent variable with a normal distribution with zero mean and unit variance. The role of the second data step is to generate a sample of $N$ responses of 10 binary Rasch items, with specified values for item parameters. We have chosen item parameters located in increasing positions on the normal scale, but other values can be chosen. We can, then, directly put the chosen value of the item parameter.

### 18.4.2. MML estimation using PROC NLMIXED

MML estimation using PROC NLMIXED is done as follows:

```sas
/* transpose */
data rasch1;set rasch; keep ind X1-X10; run;
proc sort data=rasch1 by ind; run;
proc transpose data=rasch1 out=tablable name=Question Prefix=resp by ind; run;
/* recode */
data TableRasch;set tablable;
   if Question="X1" then X1 =1; else X1 =0;
   if Question="X2" then X2 =1; else X2 =0;
   if Question="X3" then X3 =1; else X3 =0;
   if Question="X4" then X4 =1; else X4 =0;
   if Question="X5" then X5 =1; else X5 =0;
   if Question="X6" then X6 =1; else X6 =0;
```
if Question="X7" then X7 =1; else X7 =0;
if Question="X8" then X8 =1; else X8 =0;
if Question="X9" then X9 =1; else X9 =0;
if Question="X10" then X10=1; else X10=0;
run;

/* estimate */
proc nlmixed data=TableRasch;
  parms beta1=0 beta2=0 beta3=0 beta4=0 beta5=0
    beta6=0 beta7=0 beta8=0 beta9=0 beta10=0 sigma=1;
  eta=theta-(beta1*X1+beta2*X2+beta3*X3+beta4*X4+beta5*X5
                  +beta6*X6+beta7*X7+beta8*X8+beta9*X9+beta10*X10);
  expeta=exp(eta);
P=expeta/(1+expeta);
model resp1 " binomial(1,p);
random theta " normal(0,sigma*sigma) subject=ind;
run;

The sorted Rasch file is an SAS file with 11 columns, items X1 to X10 and ind the number (index) of each person. We create a new table, called OUTTABLE, with 10,000 rows and three column variables, where Question is a character variable containing the name of the corresponding item. Next, indicators variables are created. These are used by NLMIXED that does not support the CLASS statement. For an example of how to do this for polytomous items see [CHR 06].

18.4.3. MML estimation of using PROC GLIMMIX

MML estimation of the item parameters of the Rasch model can be done using PROC GLIMMIX:

proc glimmix data=TableRasch method=quad;
  class question ind;
  model resp1=question/noint dist=binomial link=logit solution cl;
  random intercept/subject=ind;
  output out=fic pred=p resid=r;
run;

18.4.4. JML estimation using PROC LOGISTIC

JML estimation of the item parameters of the Rasch model can be done using PROC LOGISTIC

proc logistic data=TableRasch;
  class question ind/PARAM=GLM;
  model resp1=ind question/noint TECH=NEWTON ABSCONV=0.0001 cl;
run;
18.4.5. **CML estimation using PROC GENMOD**

CML estimation of the item parameters of the Rasch model can be done using PROC GENMOD

```sas
proc freq data=rasch;  
   table X1*X2*X3*X4*X5*X6*X7*X8*X9*X10/sparse out-table nosprint;  
   run;  
data new; set table;  
   msscore=x1*x2*x3*x4*x5*x6*x7*x8*x9*x10;  
   if COUNT=0 then COUNT=0.0000000001;  
   run;  
ods output genmod.estimates=est;  
/* estimate */  
proc genmod data=new;  
   where 0<msscore<10;  
   class msscore;  
   model count=msscore X1 X2 X3 X4 X5 X6 X7 X8 X9 X10 / dist=binomial;  
   estimate 'beta1' X1 9 * -1 * X3 -1 X4 -1 * X6 -1 * X7 -1 * X8 -1 * X9 -1;  
   estimate 'beta2' X1 -1 X2 9 X3 -1 X4 -1 X5 -1 X6 -1 X7 -1 X8 -1 X9 -1;  
   estimate 'beta3' X1 -1 X2 -1 X3 9 X4 -1 X5 -1 X6 -1 X7 -1 X8 -1 X9 -1;  
   estimate 'beta4' X1 -1 X2 -1 X3 -1 X4 9 X5 -1 X6 -1 X7 -1 X8 -1 X9 -1;  
   estimate 'beta5' X1 -1 X2 -1 X3 -1 X4 -1 X5 9 X6 -1 X7 -1 X8 -1 X9 -1;  
   estimate 'beta6' X1 -1 X2 -1 X3 -1 X4 -1 X5 -1 X6 9 X7 -1 X8 -1 X9 -1;  
   estimate 'beta7' X1 -1 X2 -1 X3 -1 X4 -1 X5 -1 X6 -1 X7 9 X8 -1 X9 -1;  
   estimate 'beta8' X1 -1 X2 -1 X3 -1 X4 -1 X5 -1 X6 -1 X7 -1 X8 9 X9 -1;  
   estimate 'beta9' X1 -1 X2 -1 X3 -1 X4 -1 X5 -1 X6 -1 X7 -1 X8 -1 X9;  
   estimate 'beta10' X1 -1 X2 -1 X3 -1 X4 -1 X5 -1 X6 -1 X7 -1 X8 -1 X9 9;  
   run;  
/* output */  
data est;  
   set est;  
   estimate=:BetaEstimate/10; lower=:.BetaLowerCL/10; upper=:.BetaUpperCL/10;  
   keep estimate lower upper;  
   run;  
proc print data=est round modes; run;  
```

18.4.6. **JML estimation using PROC LOGISTIC**

JML estimation of the item parameters of the Rasch model can be done using PROC LOGISTIC

```sas
ods output logistic.cllparm=est5;  
proc logistic data=TableRasch;  
   class question ind/PARAM=GLM;  
   model resp(event='1')=ind question / noint TECH=NEWTON ABSFCNV=0.0001;  
   output out=icl pred=p;  
   run;  
ods rtf file='C:\file.rtf'; proc print data=est; run; ods rtf close;  
```
18.4.7. Results

Results are shown in the table below. Values of true item parameters are in the second columns. All procedures (NLMIXED, GLIMMIX, GENMOD and LOGISTIC) provide confidence interval estimates including corresponding true value.

<table>
<thead>
<tr>
<th></th>
<th>True value</th>
<th>MML NLMIXED</th>
<th>CML GENMOD</th>
<th>MML GLIMMIX</th>
<th>JML LOGISTIC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Est. lower</td>
<td>upper</td>
<td>Est. lower</td>
<td>upper</td>
</tr>
<tr>
<td>β₁</td>
<td>−1.65</td>
<td>1.60</td>
<td>1.42</td>
<td>1.78</td>
<td>−1.64</td>
</tr>
<tr>
<td>β₂</td>
<td>−1.04</td>
<td>0.99</td>
<td>0.83</td>
<td>1.15</td>
<td>−1.02</td>
</tr>
<tr>
<td>β₃</td>
<td>−0.67</td>
<td>0.75</td>
<td>0.59</td>
<td>0.90</td>
<td>−0.78</td>
</tr>
<tr>
<td>β₄</td>
<td>−0.39</td>
<td>0.32</td>
<td>0.17</td>
<td>0.47</td>
<td>−0.35</td>
</tr>
<tr>
<td>β₅</td>
<td>−0.13</td>
<td>0.03</td>
<td>−0.12</td>
<td>0.18</td>
<td>−0.07</td>
</tr>
<tr>
<td>β₆</td>
<td>0.13</td>
<td>−0.23</td>
<td>−0.38</td>
<td>−0.08</td>
<td>0.20</td>
</tr>
<tr>
<td>β₇</td>
<td>0.39</td>
<td>−0.39</td>
<td>−0.54</td>
<td>−0.23</td>
<td>0.35</td>
</tr>
<tr>
<td>β₈</td>
<td>0.67</td>
<td>−0.65</td>
<td>−0.80</td>
<td>−0.49</td>
<td>0.62</td>
</tr>
<tr>
<td>β₉</td>
<td>1.04</td>
<td>−1.06</td>
<td>−1.22</td>
<td>−0.90</td>
<td>1.03</td>
</tr>
<tr>
<td>β₁₀</td>
<td>1.65</td>
<td>−1.70</td>
<td>−1.89</td>
<td>−1.52</td>
<td>1.67</td>
</tr>
<tr>
<td>σ</td>
<td>1.0042</td>
<td>0.9251</td>
<td>1.0832</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 18.1. Comparison of item parameter estimates obtained using four different methods

The table illustrates the advantages of MML and CML estimates compared to JML estimates because the confidence interval of the JML estimates is ten times larger than the confidence of the other estimates. Furthermore, the implementation in PROC LOGISTIC as shown here, is quite time consuming.

18.5. Bibliography


[KRE 03] Kreiner S., Introduction to DIGRAM, Research report 03/10, Department of Biostatistics, University of Copenhagen, Copenhagen, 2003.


